

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-242534

(43)Date of publication of application : 08.09.2000

(51)Int.Cl.

G06F 12/00

(21)Application number : 11-045698

(71)Applicant : SHARP CORP

(22)Date of filing : 24.02.1999

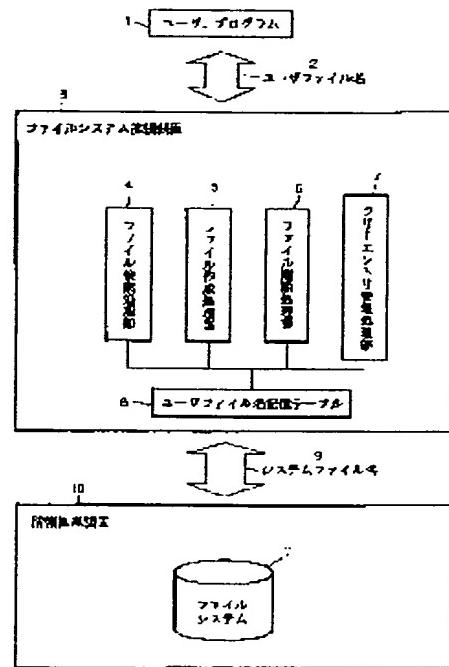
(72)Inventor : TAKIGUCHI YASUHIRO

(54) INDIRECT FILE ACCESSING METHOD, INFORMATION PROCESSOR EXECUTING SAME METHOD AND RECORDING MEDIUM WITH INDIRECT FILE ACCESS PROCESSING PROGRAM RECORDED THEREIN

(57)Abstract:

PROBLEM TO BE SOLVED: To efficiently release the limitation of a file name in an information processor having a file system that has limitation in the number of characters and a character set which can be used as a file name.

SOLUTION: This information processor comprises a user file name storage table 8 which stores a user file name 2 that does not have limitation in the number of characters and a character set to be used, a system file name converting means which converts the name 2 into a system file name 9 that has limitation in the number of characters and a character set, a file system accessing means which uses the converted system file name to access a file system 11, etc. When an existing file is retrieved, the name 2 is converted, in the table 8, into the name 9 in which includes storage position information including the name, and the file accessing means accesses the system 11.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(11) Publication number: Japanese Patent Laid-Open No. 2000-242534
(43) Date of publication of application: 08.09.2000
(51) Int. CI.: G06F 12/00

5 (21) Application number: Japanese Patent Application No. 11-045698
(22) Date of filing: 24.02.1999
(71) Applicant: SHARP CORP
(72) Inventor: TAKIGUCHI YASUHIRO

10 (54) Title of Invention: INDIRECT FILE ACCESSING METHOD, INFORMATION PROCESSOR FOR EXECUTING SAME METHOD AND RECORDING MEDIUM FOR RECORDING INDIRECT FILE ACCESS PROCESSING PROGRAM

15 **SPECIFICATION <EXCERPT>**

[0002]

[Background Art] In general, a limitation is imposed in a length of file name used in a small-scale information processor, and only the 20 file name with the limited length can be used. Moreover, a character set which can be used as a file name is limited to be alphabet characters and numerals, and the alphabet characters cannot have distinction of capital letters and small letters. In order to avoid the length limitation of the file name in the filing system, a 25 method has been suggested to calculate a hash value which uses an original file name as a key, and to use this value instead of the original file name as a fixed-length conversion file name, but it has been difficult to efficiently manage the files. Furthermore, in order to avoid the length limitation of the file name, another method has 30 been suggested to divide an area for storing file management information into a fixed-length file name storage area and a constant information area for storing extra parts of the fixed length

file name and to manage the files, but it has been also difficult efficiently manage the files.

[0003]

[Problem(s) to be Solved by the Invention] When the character set

5 which can be used as the file name in the information processor has a limit, it is not possible to create a file which has alphabetic characters beyond the range of the character set. Moreover, when the number of alphabetic characters which can be used as the file name has a limit, only the file name within the limit can be used, and
10 names in which limited parts are the same but the other parts beyond the limit are different each other are considered the identical name.

[0004] The method by which the hash value of the original file name is used as a converted name instead of the original file name

15 in order to extend the length limitation of the file name has a problem that the calculation of the hash value is complicated which results in complicated management in order to avoid collision of the converted file names when the calculated hash values are overlapped, so that it has been difficult to efficiently manage the
20 files. Regarding the management method by which the area for storing file management information is divided into the fixed-length file name storage area and the constant information area for storing extra parts of the fixed length file name and to manage the files, the method needs additional managements for managing when there
25 are files whose fixed-length file name parts are overlapped, and for managing complicated parts in the file name which is beyond the limit, so that it has been difficult to efficiently manage the files.

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2000-242534

(P2000-242534A)

(43)公開日 平成12年9月8日(2000.9.8)

(51)Int.Cl.⁷

G 0 6 F 12/00

識別記号

5 2 0

F I

G 0 6 F 12/00

テ-マコード*(参考)

5 2 0 G 5 B 0 8 2

審査請求 未請求 請求項の数9 O L (全12頁)

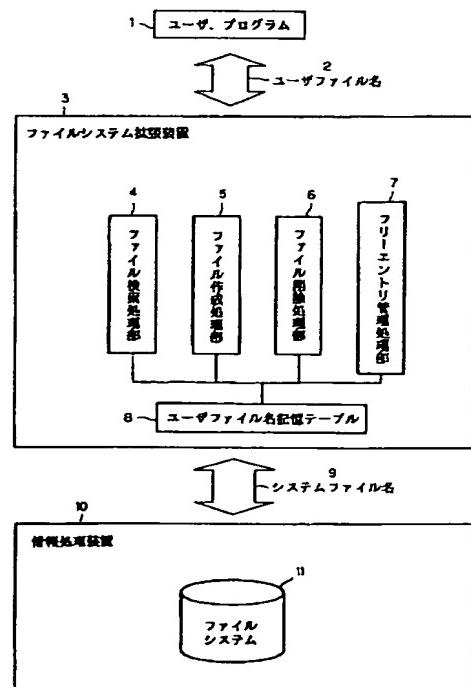
(21)出願番号	特願平11-45698	(71)出願人	000005049 シャープ株式会社 大阪府大阪市阿倍野区長池町22番22号
(22)出願日	平成11年2月24日(1999.2.24)	(72)発明者	滝口 康弘 大阪府大阪市阿倍野区長池町22番22号 シ ヤープ株式会社内

(54)【発明の名称】 間接ファイルアクセス方法、該方法を実施する情報処理装置及び間接ファイルアクセス処理プログラムを記録した記録媒体

(57)【要約】

【目的】 ファイルの名前として用いることのできる文字数及び文字セットに制限があるファイルシステムを有する情報処理装置において、ファイル名の前記制限を効率的に解除する。

【構成】 情報処理装置は、使用する文字数及び文字セットに制限がないユーザファイル名2を記憶するユーザファイル名記憶テーブル8、ユーザファイル名2を文字数及び文字セットに制限があるシステムファイル名9に変換するシステムファイル名変換手段、変換されたシステムファイル名を用いてファイルシステム11にアクセスするファイルシステムアクセス手段等からなる。既存のファイルを検索する場合、ユーザファイル名2をユーザファイル名記憶テーブル8においてそのユーザファイル名2が記憶されている記憶位置情報を含むシステムファイル名9に変換し、ファイルシステムアクセス手段によって、ファイルシステム11にアクセスする。



【特許請求の範囲】

【請求項1】 ファイル名を用いてファイルを管理するファイルシステムにおける間接ファイルアクセス方法において、
ユーザやプログラムが用いるユーザファイル名を記憶するユーザファイル名記憶手段を用意し、
ユーザやプログラムが用いる任意のユーザファイル名を、前記ユーザファイル名記憶手段においてそのユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に変換し、
該変換されたシステムファイル名をファイル名として前記ファイルシステムにアクセスし、
また、システムファイル名から前記ユーザファイル名記憶手段における記憶位置情報を取り出し、前記ユーザファイル名記憶手段を参照してユーザファイル名に変換することを特徴とする間接ファイルアクセス方法。

【請求項2】 ファイル名を用いてファイルを管理するファイルシステムを有する情報処理装置において、
ユーザやプログラムが用いるユーザファイル名を記憶するユーザファイル名記憶手段と、
ユーザやプログラムが用いる任意のユーザファイル名に対して、前記ユーザファイル名記憶手段においてそのユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に変換するシステムファイル名変換手段と、
該システムファイル名変換手段によって変換されたシステムファイル名をファイル名として前記ファイルシステムにアクセスするファイルシステムアクセス手段と、
システムファイル名から前記ユーザファイル名記憶手段における記憶位置情報を取り出し、前記ユーザファイル名記憶手段を参照してユーザファイル名に変換するユーザファイル名変換手段と、を備えたことを特徴とする情報処理装置。

【請求項3】 ファイル名を用いてファイルを管理するファイルシステムを有し、該ファイルシステムにおけるファイル名として用いることができる文字セットに制約があり、かつユーザやプログラムが用いるユーザファイル名として前記ファイルシステムにおけるファイル名として用いることができる文字セット以外の文字セットを用いることが可能な情報処理装置において、

前記ユーザファイル名を前記ファイルシステムが用いるシステムファイル名と対応づけて記憶するユーザファイル名記憶手段と、
前記ユーザファイル名を前記システムファイル名として用いることができる文字セットの制約に従ったシステムファイル名に変換するシステムファイル名変換手段と、
該システムファイル名変換手段によって変換されたシステムファイル名をファイル名として前記ファイルシステムにアクセスするファイルシステムアクセス手段と、
システムファイル名を前記ユーザファイル名記憶手段を

2 参照してユーザファイル名に変換するユーザファイル名変換手段と、を備えたことを特徴とする情報処理装置。

【請求項4】 請求項3記載の情報処理装置であって、前記システムファイル名変換手段は、任意のユーザファイル名に対して、前記ユーザファイル名記憶手段において前記ユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に変換し、
前記ユーザファイル名変換手段は、システムファイル名から前記ユーザファイル名記憶手段における記憶位置情報を取り出し、前記ユーザファイル名記憶手段を参照してユーザファイル名に変換することを特徴とする情報処理装置。

【請求項5】 請求項3または4記載の情報処理装置であって、前記システムファイル名変換手段は、前記ユーザファイル名が前記ファイルシステムにおいてファイル名として用いることができる文字セットの制約に従っている場合には、システムファイル名としてユーザファイル名をそのまま用いることを特徴とする情報処理装置。

【請求項6】 ファイル名を用いてファイルを管理するファイルシステムを有し、該ファイルシステムにおけるファイル名の長さに制約があり、かつユーザやプログラムが用いるユーザファイル名として前記ファイルシステムにおけるファイル名の長さの制限を超えた長さのファイル名を用いることが可能な情報処理装置において、前記ユーザファイル名を前記ファイルシステムが用いるシステムファイル名と対応づけて記憶するユーザファイル名記憶手段と、
任意のユーザファイル名に対して、前記ユーザファイル名記憶手段においてユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に、前記ファイルシステムにおけるファイル名の長さの制限に従って変換するシステムファイル名変換手段と、

該システムファイル名変換手段によって変換されたシステムファイル名をファイル名として前記ファイルシステムにアクセスするファイルシステムアクセス手段と、
システムファイル名から前記ユーザファイル名記憶手段における記憶位置情報を取り出し、前記ユーザファイル名記憶手段を参照してユーザファイル名に変換するユーザファイル名変換手段と、を備えることを特徴とする情報処理装置。

【請求項7】 請求項6記載の情報処理装置であって、前記システムファイル名変換手段は、前記ユーザファイル名の長さが前記ファイルシステムにおいてファイル名として用いることができる長さの制限に従っている場合には、システムファイル名として前記ユーザファイル名をそのまま用いることを特徴とする情報処理装置。

【請求項8】 請求項2乃至7のいずれかに記載の情報処理装置であって、
前記システムファイル名変換手段によって変換されるシステムファイル名の特定の位置に特定の文字列が含ま

れ、前記ユーザファイル名変換手段において処理されるシステムファイル名の特定の位置に特定の文字列が含まれる場合には、ユーザファイル名として前記システムファイル名をそのまま用いることを特徴とする情報処理装置。

【請求項9】 コンピュータに、
ユーザやプログラムが使用するユーザファイル名を記憶するユーザファイル名記憶機能と、
前記ユーザファイル名記憶機能を用いて、ユーザやプログラムが使用する任意のユーザファイル名をそのユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に変換するシステムファイル名変換機能と、
前記システムファイル名変換機能を用いて、変換されたシステムファイル名をファイル名としてファイルシステムにアクセスし、そのシステムファイル名から記憶位置情報を取り出し、ユーザファイル名に変換するユーザファイル名変換機能と、を実行させるためのプログラムを記録した記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、ファイルの名前として用いることが可能な文字数に制限がある情報処理装置やファイルの名前として用いることが可能な文字セットに制約がある情報処理装置において、これらの制限や制約を解除し、十分長い文字数で多様な文字セットを用いたユーザファイル名を使用することが可能な間接ファイルアクセス方法及び該方法を実施する情報処理装置並びに間接ファイルアクセス処理プログラムを記録した記録媒体に関する。

【0002】

【従来の技術】 一般的に小規模の情報処理装置で利用できるファイルの名前として用いることができる長さには制限が設けられており、その制限に収まるファイル名しか使用する事はできない。また、ファイル名として用いることが可能な文字セットとしてはアルファベット文字と数字に限定されたり、アルファベット文字については、大文字と小文字の区別を付けられないといった制約がある。このファイルシステムのファイル名の長さの制限を回避する手段としては、元のファイル名をキーとしたハッシュ値を求め、この値を固定長の変換ファイル名として元のファイル名の代わりに用いる方法が提案されているが効率的なファイルの管理が難しかった。同様にファイル名の長さの制約を回避する手段として、ファイルの管理情報を格納する領域を固定長のファイル名格納領域とそれを越える部分を格納する継続情報領域とに分けて管理する方法が提案されているが、この場合も効率的なファイルの管理が難しかった。

【0003】

【発明が解決しようとする課題】 情報処理装置で利用で

きるファイルの名前として使用することができる文字セットに制約がある場合には、その文字セットの範囲を越えた文字を用いたファイルを作成することはできない。又、ファイル名として利用できる文字数に制限がある場合には、その制限に収まるファイル名しか使用することはできず、それを越えた部分が異なるファイルは同一の名前として処理が行なわれてしまう。

【0004】 ファイル名の長さの制限を拡張するために元のファイル名のハッシュ値を変換ファイル名とし、元のファイルの代わりとして用いる方法はハッシュ値の計算が複雑であり、計算したハッシュ値が衝突した場合の変換後のファイル名の衝突を回避するには更に複雑な管理が要され効率的なファイル管理が難しかった。ファイルの管理情報を格納する領域を固定長のファイル名格納領域とそれを越える部分を格納する継続情報領域とに分けて管理する方法については、固定長のファイル名部分が衝突するファイルがある場合の管理方法や固定長を越えた部分の複雑なファイル名部分の管理が別途必要となり、効率的なファイル管理が難しかった。

【0005】 本発明は、情報処理装置が備えた既存のファイルシステムの仕組みには手を加えず、従来のファイルシステムが持っていた各種制約、例えば利用できる文字セットの制約やファイル名として用いることができる文字数の制約等、を効率良く取り除き、ユーザインターフェースを大幅に改善することができる間接ファイルアクセス方法、該方法を実施する情報処理装置及び間接ファイルアクセスプログラムを記録した記録媒体を提供することにある。

【0006】

【課題を解決するための手段】 本発明は、ファイル名として用いる事ができる文字セットやファイル名として用いることができるファイル長に制限があるファイルシステムを有する情報処理装置において、ファイル名として用いることができる文字セットとファイル名の長さの制約の無いファイル名を記憶するユーザファイル名記憶テーブルを用意し、ファイルのアクセスを行なう場合にはそのテーブルを検索し、そのファイル名が記憶されたテーブル内の位置情報を基にした実ファイル名に変換することで、上記の制約を取り除くものである。

【0007】 請求項1の発明は、ファイル名を用いてファイルを管理するファイルシステムにおける間接ファイルアクセス方法において、ユーザやプログラムが用いるユーザファイル名を記憶するユーザファイル名記憶手段を用意し、ユーザやプログラムが用いる任意のユーザファイル名を、前記ユーザファイル名記憶手段においてそのユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に変換し、該変換されたシステムファイル名をファイル名として前記ファイルシステムにアクセスし、また、システムファイル名から前記ユーザファイル名記憶手段における記憶位置情報を取り出し、

前記ユーザファイル名記憶手段を参照してユーザファイル名に変換することを特徴とする間接ファイルアクセス方法に関する。

【0008】請求項2の発明は、ファイル名を用いてファイルを管理するファイルシステムを有する情報処理装置において、ユーザやプログラムが用いるユーザファイル名を記憶するユーザファイル名記憶手段と、ユーザやプログラムが用いる任意のユーザファイル名に対して、前記ユーザファイル名記憶手段においてそのユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に変換するシステムファイル名変換手段と、該システムファイル名変換手段によって変換されたシステムファイル名をファイル名として前記ファイルシステムにアクセスするファイルシステムアクセス手段と、システムファイル名から前記ユーザファイル名記憶手段における記憶位置情報を取り出し、前記ユーザファイル名記憶手段を参照してユーザファイル名に変換するユーザファイル名変換手段とを備えたことを特徴とする情報処理装置に関する。

【0009】請求項3の発明は、ファイル名を用いてファイルを管理するファイルシステムを有し、該ファイルシステムにおけるファイル名として用いることができる文字セットに制約があり、かつユーザやプログラムが用いるユーザファイル名として前記ファイルシステムにおけるファイル名として用いることができる文字セット以外の文字セットを用いることが可能な情報処理装置において、前記ユーザファイル名を前記ファイルシステムが用いるシステムファイル名と対応づけて記憶するユーザファイル名記憶手段と、前記ユーザファイル名を前記システムファイル名として用いることができる文字セットの制約に従ったシステムファイル名に変換するシステムファイル名変換手段と、該システムファイル名変換手段によって変換されたシステムファイル名をファイル名として前記ファイルシステムにアクセスするファイルシステムアクセス手段と、システムファイル名を前記ユーザファイル名記憶手段を参照してユーザファイル名に変換するユーザファイル名変換手段とを備えたこと特徴とする情報処理装置に関する。

【0010】請求項4の発明は、請求項3記載の発明において、前記システムファイル名変換手段は、任意のユーザファイル名に対して、前記ユーザファイル名記憶手段において前記ユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に変換し、前記ユーザファイル名変換手段は、システムファイル名から前記ユーザファイル名記憶手段における記憶位置情報を取り出し、前記ユーザファイル名記憶手段を参照してユーザファイル名に変換することを特徴とする情報処理装置に関する。

【0011】請求項5の発明は、請求項3または4記載の発明において、前記システムファイル名変換手段は、

前記ユーザファイル名が前記ファイルシステムにおいてファイル名として用いることができる文字セットの制約に従っている場合には、システムファイル名としてユーザファイル名をそのまま用いることを特徴とする情報処理装置に関する。

【0012】請求項6の発明は、ファイル名を用いてファイルを管理するファイルシステムを有し、該ファイルシステムにおけるファイル名の長さに制約があり、かつユーザやプログラムが用いるユーザファイル名として前記ファイルシステムにおけるファイル名の長さの制限を超えた長さのファイル名を用いることが可能な情報処理装置において、前記ユーザファイル名を前記ファイルシステムが用いるシステムファイル名と対応づけて記憶するユーザファイル名記憶手段と、任意のユーザファイル名に対して、前記ユーザファイル名記憶手段においてユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に、前記ファイルシステムにおけるファイル名の長さの制限に従って変換するシステムファイル名変換手段と、該システムファイル名変換手段によって変換されたシステムファイル名をファイル名として前記ファイルシステムにアクセスするファイルシステムアクセス手段と、システムファイル名から前記ユーザファイル名記憶手段における記憶位置情報を取り出し、前記ユーザファイル名記憶手段を参照してユーザファイル名に変換するユーザファイル名変換手段とを備えることを特徴とする情報処理装置に関する。

【0013】請求項7の発明は、請求項6記載の発明において、前記システムファイル名変換手段は、前記ユーザファイル名の長さが前記ファイルシステムにおいてファイル名として用いることができる長さの制限に従っている場合には、システムファイル名として前記ユーザファイル名をそのまま用いることを特徴とする情報処理装置に関する。

【0014】請求項8の発明は、請求項2乃至7のいずれかに記載の発明において、前記システムファイル名変換手段によって変換されるシステムファイル名の特定の位置に特定の文字列が含まれ、前記ユーザファイル名変換手段において処理されるシステムファイル名の特定の位置に特定の文字列が含まれる場合には、ユーザファイル名として前記システムファイル名をそのまま用いることを特徴とする情報処理装置に関する。

【0015】請求項9の発明は、コンピュータに、ユーザやプログラムが使用するユーザファイル名を記憶するユーザファイル名記憶機能と、前記ユーザファイル名記憶機能を用いて、ユーザやプログラムが使用する任意のユーザファイル名をそのユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に変換するシステムファイル名変換機能と、前記システムファイル名変換機能を用いて、変換されたシステムファイル名をファイル名としてファイルシステムにアクセスし、その

システムファイル名から記憶位置情報を取り出し、ユーザファイル名に変換するユーザファイル名変換機能と、を実行させるためのプログラムを記録した記録媒体に関する。

【0016】(作用) 本発明によって情報処理装置のファイルシステムで用いることが出来るファイル名として使える文字セットに対する制限が無くなり、ファイル名として日本語を含む文字をファイル名として用いることが可能となる。更にファイルに付けることができる名前の長さについての制約を事実上無くすことが可能となり、任意の長さのファイル名を付けることが可能となる。

【0017】

【発明の実施の形態】 本発明の実施例を図を用いて説明する。図1は、間接ファイルアクセス方法を実施するためのファイルシステム拡張装置と情報処理装置、並びにユーザとプログラムの関係を示したものである。1はユーザや情報処理装置上のプログラムで、この部分で用いられる変換前のユーザファイル名2は使用する文字セットに制約がなく、ファイル名として用いることができる文字の長さについての制約もないファイル名である。ファイルシステム拡張装置3は、ファイル検索処理部4、ファイル作成処理部5、ファイル削除処理部6、フリーエントリ管理処理部7、並びにファイル名記憶テーブル8とから構成されている。このファイルシステム拡張装置3で変換前のユーザファイル名2は名前の変換が行なわれて、変換後のシステムファイル名9となり、ファイル名に用いることができる文字セットやファイル名の長さの制約がある情報処理装置10に入力される。

【0018】(ユーザファイル名記憶テーブルの実施例) 図2は、ユーザファイル名記憶テーブル8の実施例を示したものである。図2の場合には、エントリとして1～Zのファイル記憶テーブルがあり、総数でZ個のファイルが存在していることを示している。なお、この実施例における情報処理装置10で利用することができるファイル名については利用可能な文字セットはアルファベット大文字と数値を合わせた36文字とし、ファイル名としてサフィックスを付けることが可能で、サフィックスの前にはコンマ「,」を付けるものとする。また、ファイル名の長さについては、サフィックスより前の部分は長さ8文字、サフィックス部分については3文字に制約されているものとする。

【0019】 個々のエントリは、エントリのインデックス12、使用／未使用フラグ13、変換前ファイル名14、フリーエントリリスト15から構成されている。使用／未使用フラグ13はこのエントリが使用されているか否かを示すフラグで、例えばこの値が1であれば使用されているエントリであることを示し、この値が0であれば未使用のエントリであることを示す。変換前ファイル名14は使用できる文字セットについての制約が取り

払われ、ファイル名として用いることが出来る文字数の制約が拡張された変換前のユーザファイル名2を格納する領域である。この図の場合このファイル名の長さは最大128文字となっており、制限が大幅に少なくなっている。この領域の大きさは事実上ファイル名の長さの制約が問題無い大きさに設定される。次のフリーエントリリスト15はエントリのうちで未使用のものを繋ぐ機能を有し、空きエントリの再利用を効率的に行なうためのものである。図2の場合、エントリインデックス3のエントリは空きエントリであるが、その次の空きエントリのインデックスは5、その次の空きエントリのインデックスはXであることを示している。このようなユーザファイル名記憶テーブル8の個々のエントリに対して、その中に格納されたユーザファイル名に対するシステムファイル名は、テーブル内の位置情報を基にしたものである。この名前は例えば、ファイル名記憶テーブルのインデックス値で済ませることができる。その場合インデックス4のシステムファイル名は00000004となる。00000004はインデックス値4を8文字で表現した文字列である。また、このファイルがファイル名が変換されていることを示すために特殊なサフィックスを追加して例えば00000004.CNVといった名前にするとファイル名が変換されたファイルであることがサフィックスをみることで分かる。

【0020】(即存のファイルを検索する場合の実施例) 図3は、即存のファイルを検索する場合の処理を示すフローチャートである。即存のファイルを検索する場合の処理を図2を基にして、図3のフローチャートを用いて説明する。プログラムで指定された変換前のユーザファイル名2はファイルシステム拡張装置3に入力され、そこでファイル検索処理部4にてファイルシステム1-1で用いる実体ファイルの名前に変換される。この処理はユーザファイル名記憶テーブル8の先頭エントリの内容を取り出すことから始まる(S1)。この時、ユーザファイル名記憶テーブル8のインデックスを記憶しておく(S2)。次に、取り出したエントリの中の使用／未使用フラグ13を取り出し(S3)、このエントリが現在使用中であるかどうかをチェックする(S4)。もし、未使用であればユーザファイル名記憶テーブル8内の最後のエントリであるかどうかをチェックし(S5)、最後のエントリであれば、ユーザファイル名記憶テーブル8には検索しているファイルは存在しないことになり処理を終了する。もし、ステップ5において、最後のエントリでなければユーザファイル名記憶テーブル8の次のエントリの内容を取り出し(S6)、この時記憶していた記憶テーブルのインデックスをインクリメントし(S7)、ステップ5に戻る。もし、ステップ4の処理にて、このエントリが使用中であった場合、ユーザファイル名記憶テーブル8から変換前ファイル名14を取り出して(S8)、検索しているファイル名であるか

どうかをチェックする（S 9）。もし、ファイル名が異なれば前記ステップ5の処理から処理を継続する。もし、ファイル名が同じであれば、検索ファイルの変換後のファイル名9は、このエントリのインデックスを名前とするファイルであるとする（S 10）。

【0021】（新規のファイルを作成する場合の実施例）図4は、新規ファイルを作成する場合の処理を説明するためのユーザファイル名記憶テーブル8の構成を示し、図5は、その処理を示すフローチャートである。新規ファイルを作成する場合の処理を図4と、図5のフローチャートに基づいて説明する。まず、図5のフローチャートのように、今から作るファイルが既に存在しているかどうかをチェックするため既存ファイルを検索し（S 21）、同名のファイルが存在するかどうかチェックする（S 22）。ここで、同名のファイルが存在すれば、既に同じ名前のファイルが存在しているのであるから、新規のファイル作成を失敗させる。もし、ステップ22において、同名のファイルが存在しない場合には、フリーエントリリストが存在するかどうかを、フリーエントリリスト15を参照することによりチェックする（S 23）。もし、フリーエントリが存在しない場合には、ユーザファイル名記憶テーブル8内のエントリは全て現在使用中であることを意味し、ユーザファイル名記憶テーブル8自身のテーブルサイズを拡張する必要がある。この拡張は、ユーザファイル名記憶テーブル8の最後にエントリを1つ追加することによって行なう（S 24）。こうして作成した新しいエントリに対して使用／未使用フラグ13をセットにし（S 25）、変換前ファイル名14の領域に新規に作成するファイル名を登録する（S 26）。変換後のファイル名はこの新しいエントリのインデックス番号であるため、その変換後ファイル名でファイルを作成する（S 30）。

【0022】もし、ステップ23においてフリーエントリが存在する場合には、かつては利用されていたが、ファイルが削除されたためにフリーになって未使用状態になったエントリが存在すると判断される。このようなエントリは、図2の場合であれば、インデックス3、インデックス5、インデックスX…といったエントリである。このエントリはフリーリストの先頭16によって指示されている。図2の場合にはエントリ3が先頭エントリである。このエントリを新規に作成するファイルのために再利用を行なう。まず、使用／未使用フラグ13をセットし（S 27）、このエントリが使用中であることを示す設定を行なう。次に変換前ファイル名14の部分に新規に作成するファイル名を登録し（S 28）、フリーリストの先頭16をインデックス3のエントリの次のエントリ5に修正する（S 29）。こうして確保されたエントリ3が新規に作られるファイルの変換後のファイル名となる（S 30）。

【0023】（既存のファイルを削除する場合の実施

例）図6は、既存のファイルを削除する場合の処理を説明するためのファイル名記憶テーブル8の構成を示し、図7は、その処理を示すフローチャートである。即存のファイルを削除する場合の処理を図6と図7のフローチャートを用いて説明する。最初に、即存ファイルの検索処理を行ない（S 41）、これから削除するファイルがユーザファイル名記憶テーブル8に登録されているかどうかをチェックする（S 42）。もし、ファイル名記憶テーブル8に登録されていなければ、削除するファイルは存在しないことになり処理を終了する。もし、ファイル名記憶テーブル8に登録されていれば、削除するファイルのユーザファイル名記憶テーブル8内でのインデックスと変換後のファイル名がステップ41の処理によって分かる。ユーザファイル名記憶テーブル8の当該エントリを再利用できるようにするために、使用／未使用フラグ13をリセットし（S 43）、当該エントリのフリーエントリリスト15の値をフリーリストの先頭16の値にし（S 44）、フリーエントリの先頭16の値は当該エントリを指示するようにする（S 45）。その後で変換後ファイルでファイルを削除する（S 46）。

【0024】（使われている文字セットがファイルシステムで用いることが可能な文字セットの制約に収まっていてファイル名の変換を行なわないで済む場合の実施例）図8は、使われている文字セットがファイルシステム11で用いることが可能な文字セットの制約に収まっていてファイル名の変換を行なわないで済む場合の処理を示すフローチャートである。ユーザやプログラムが入力するユーザファイル名で用いられている文字セットが情報処理装置10のファイルシステム11で用いることができる文字セットの制約を満たしている場合には、ユーザファイル名記憶テーブル8を介すことなく、直接情報処理装置に対してファイル名を入力することでファイル名の変換を行なわずに済ませることが可能である。この場合には、図8のように、まず最初にユーザやプログラムが入力するユーザファイル名2で用いられている文字セットが情報処理装置で用いる事が可能な文字セットのみで構成されているかどうかを判別する。もし、利用可能な文字セットのみでファイル名が構成されている場合には、ユーザファイル名記憶テーブル8を介したファイル検索処理やファイル作成処理、ファイル削除処理、フリーエントリ管理処理を行なう事なく直接のファイル処理を行なう。また、利用不可能な文字を含んだ文字セットでファイルが構成されている場合には、ユーザファイル名記憶テーブル8を介して、ファイル名の変換を行うファイル処理を行う。

【0025】（ファイル名の特定の位置に特定の文字列が使用されていて、ファイル名の変換を行なわない場合の実施例）図9は、ファイル名の特定の位置に特定の文字列が使用されていて、ファイル名の変換を行なわない場合の処理を示すフローチャートである。ユーザやプロ

グラムが入力するユーザファイル名2において特定の位置に特定の文字列が使用されている場合、前述のようなファイル変換を行なうことなく、直接情報処理装置に対してファイル名を入力することでファイル名の変換を行なわずに済ませることも可能である。実施例としては図9のようにユーザファイル名のサフィックスに特定の文字列（例えば、r a w）が指定されている場合、ユーザファイル名記憶テーブル8を介したファイル検索処理やファイル作成処理、ファイル削除処理、フリーエントリ管理処理を行なうことなく直接のファイル処理を行う。

【0026】（間接ファイルアクセス処理プログラムを記録した記録媒体の実施例）以上の各実施例によって説明した間接ファイルアクセス方法は、図1に示したシステムからなる情報処理装置でそれを実行するプログラムによって実現される。そして、間接ファイルアクセス処理を実行するプログラムは、ユーザやプログラムが使用するユーザファイル名を記憶するユーザファイル名記憶機能と、前記ユーザファイル名記憶機能を用いて、ユーザやプログラムが使用する任意のユーザファイル名をそのユーザファイル名が記憶されている記憶位置情報を含むシステムファイル名に変換するシステムファイル名変換機能と、前記システムファイル名変換機能を用いて、変換されたシステムファイル名をファイル名としてファイルシステムにアクセスし、そのシステムファイル名から記憶位置情報を取り出し、ユーザファイル名に変換するユーザファイル名変換機能とを実行させるためのプログラムで、コンピュータで読み取り可能な記録媒体に格納されている。この記録媒体は、本体と分離可能に構成されるプログラムメディアであってもよいし、本体に備えられている記憶手段そのものであってもよい。

【0027】このプログラムメディアは、磁気テープやカセットテープ等のテープ系、フロッピー（登録商標）ディスクやハードディスク等の磁気ディスクやCD-ROM, MO, MD, DVD等の光ディスク系のディスク系、ICカード、光カード等のカード系、あるいはマスクROM, EPROM, EEPROM, フラッシュROM等による半導体メモリを含めた固体的にプログラムを担持する媒体であってもよい。

【0028】また、情報処理装置に外部の通信ネットワークとの接続が可能な通信手段を備えている場合には、この通信手段を介して通信ネットワークからプログラムをダウンロードするように流動的にプログラムを担持する媒体であってもよい。このように通信ネットワークからプログラムをダウンロードする場合には、そのダウンロード用プログラムは予め本体装置に格納しておくか、あるいは別な記録媒体からインストールされるものであってもよい。なお、記録媒体に格納されている内容とし

てはプログラムに限定されず、データであってもよい。

【0029】

【発明の効果】以上のように、本発明によれば、ファイル名として使える文字セットの制約やファイル名の長さによる制約があるファイルシステムを使用する情報処理装置において、用いることが出来る文字セットに対する制限が無くなり、ファイル名として日本語を含む文字を用いることが可能となり、ファイルに付けることができる名前の長さについての制約が事実上なくなり任意の長さのファイル名を付けることが可能となる。また、情報処理装置が備えているファイルシステムを変更することなしに、簡単なハードウェア及びソフトウェア構成で効率的にファイル名の制約を解除することができる。

【図面の簡単な説明】

【図1】間接ファイルアクセス方法を実施するためのファイルシステム拡張装置と情報処理装置、及びユーザとプログラムの関係を示す図である。

【図2】ユーザファイル名記憶テーブルの構造を示す図である。

【図3】既存のファイルを検索する場合の処理を説明するためのフローチャートである。

【図4】新規のファイルを作成する場合の処理を説明するためのユーザファイル名記憶テーブルの構造を示す図である。

【図5】新規のファイルを作成する場合の処理を説明するためのフローチャートである。

【図6】既存のファイルを削除する場合の処理を説明するためのユーザファイル名記憶テーブルの構造を示す図である。

【図7】既存のファイルを削除する場合の処理を説明するためのフローチャートである。

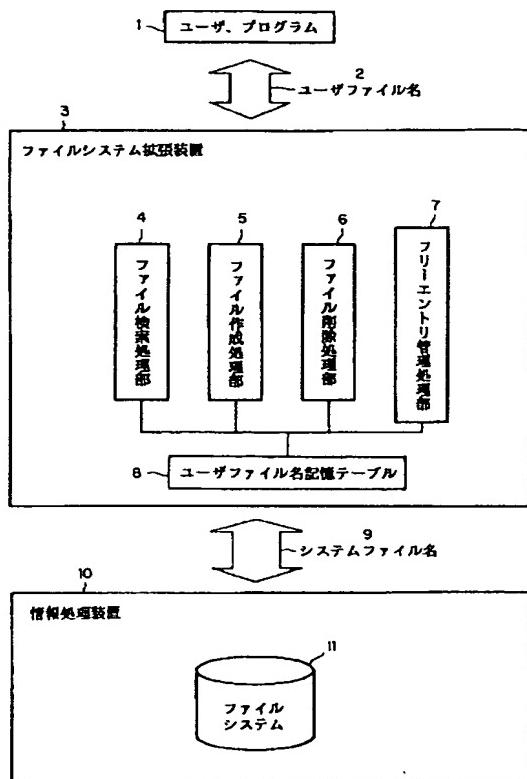
【図8】使われている文字セットがファイルシステムで用いることが可能な文字セットの制約に収まっていて、ファイル名の変換を行なわないですむ場合の処理を説明するためのフローチャートである。

【図9】ファイル名の特定の位置に特定の文字列が使用されていて、ファイル名の変換を行なわないですむ場合の処理を説明するためのフローチャートである。

【符号の説明】

1…ユーザ、プログラム、2…ユーザファイル名、3…ファイルシステム拡張装置、4…ファイル検索処理部、5…ファイル作成処理部、6…ファイル削除処理部、7…フリーエントリ管理処理部、8…ファイル名記憶テーブル、9…システムファイル名、10…情報処理装置、11…ファイルシステム、12…エントリのインデックス、13…使用／未使用フラグ、14…変換前ファイル名、15…フリーエントリスト、16…フリーリストの先頭、17…次のフリーリストのインデックス。

【図1】



【図4】

【図4】

13 14 15
1 1 XXXXXXXXXX 0
2 1 YYYYYYYYYYYY 0
3 1 新規作成したファイル名 0
4 1 UUUUUUUU 0
5 0 X
X . .
n 1 NNNNNNNN 0
. . .
z 1 ZZZZZZZZZZ 0

16 フリー リストの先頭

使用中のフラグを 2 セットする
新規作成した ファイルの置換前の ファイル名

フリーリストの 先頭を修正する
フリーでなくなるため、
インデックスを リセットする

128 バイト

【図2】

12 13 14 15
1 1 XXXXXXXXXX 0
2 1 YYYYYYYYYYYY 0
3 0 5
4 1 UUUUUUUU 0
5 0 X
X . .
n 1 NNNNNNNN 0
. . .
z 1 ZZZZZZZZZZ 0

16 フリー リストの先頭

128 バイト

【図6】

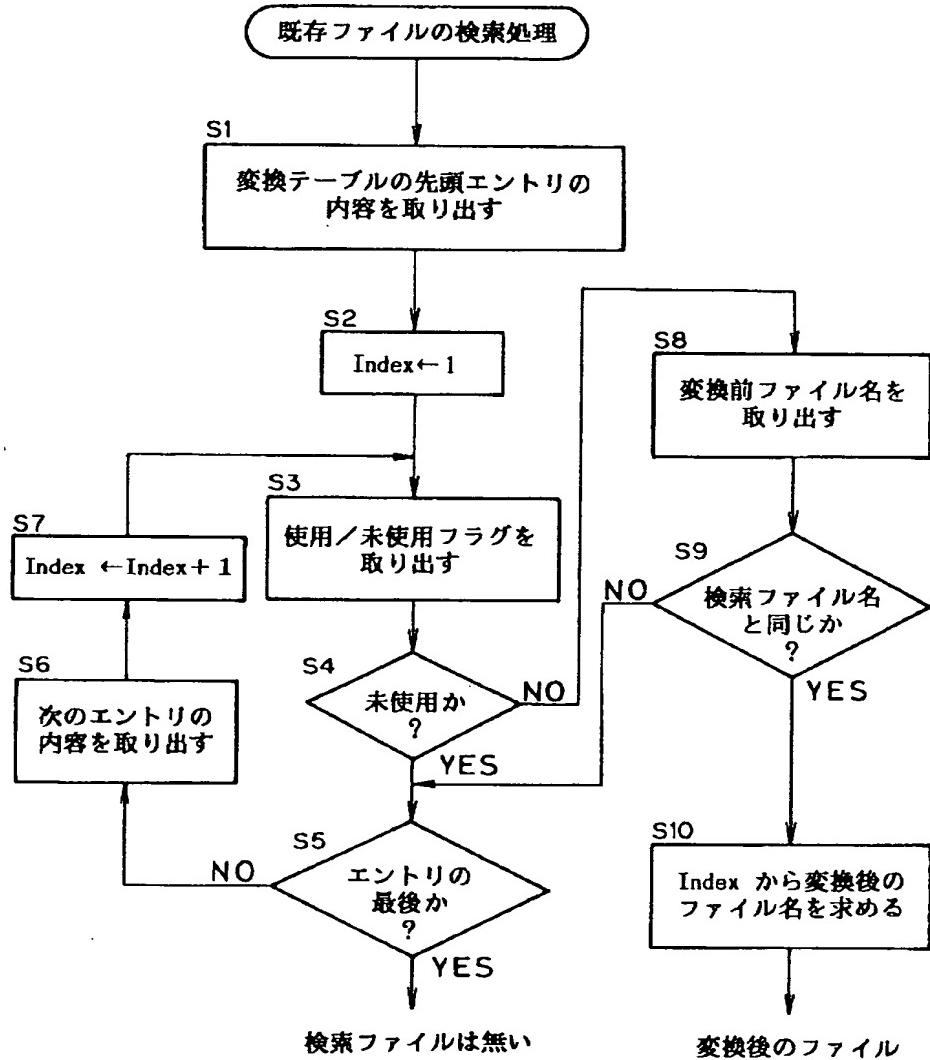
13 14 15
1 1 XXXXXXXXXX 0
2 1 YYYYYYYYYYYY 0
3 0 5
4 1 UUUUUUUU 0
5 0 X
X . .
n 0 NNNNNNNN 0
. . .
z 1 ZZZZZZZZZZ 0

16 フリー リストの先頭

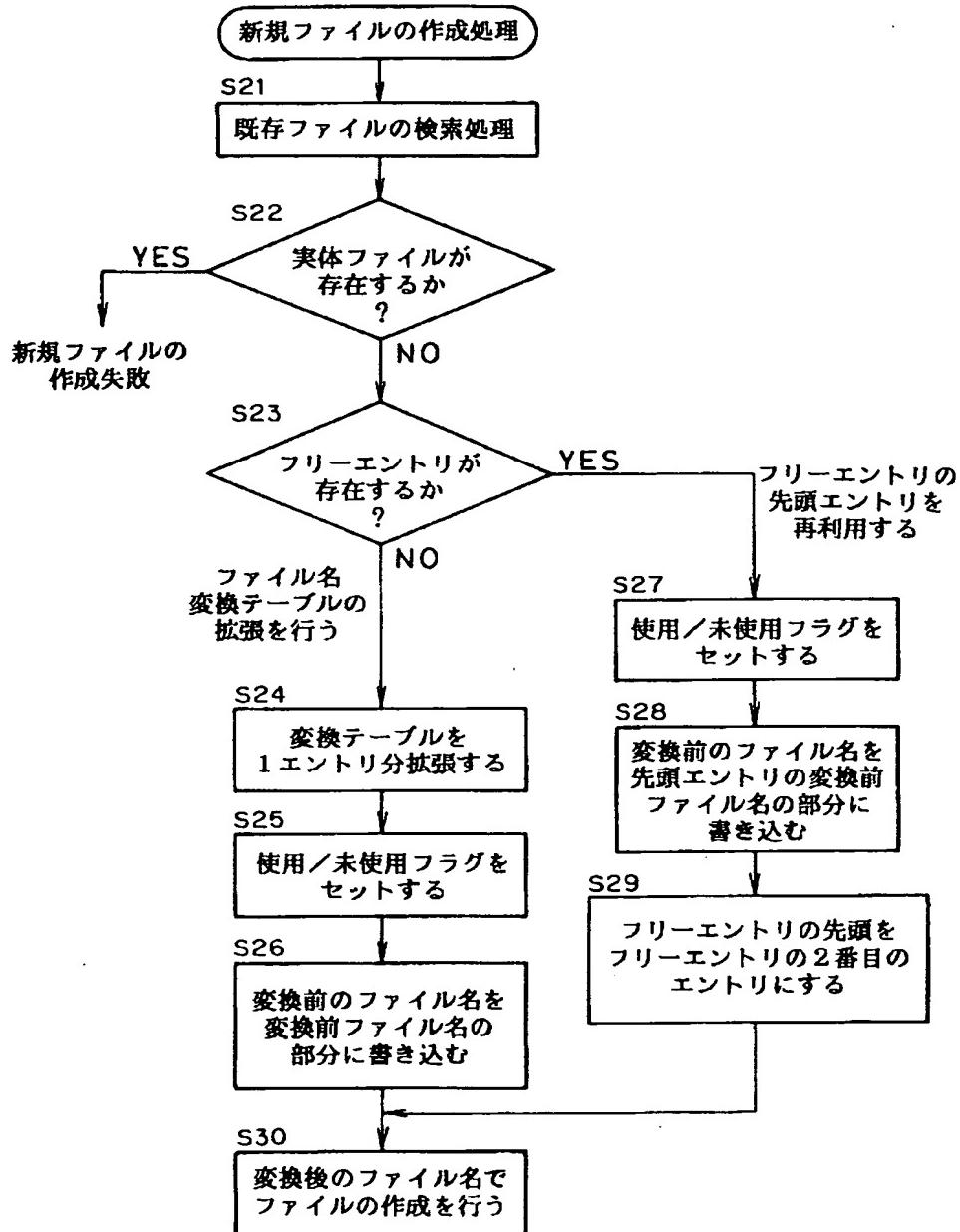
削除するため 0 にする
削除する ファイル名

128 バイト

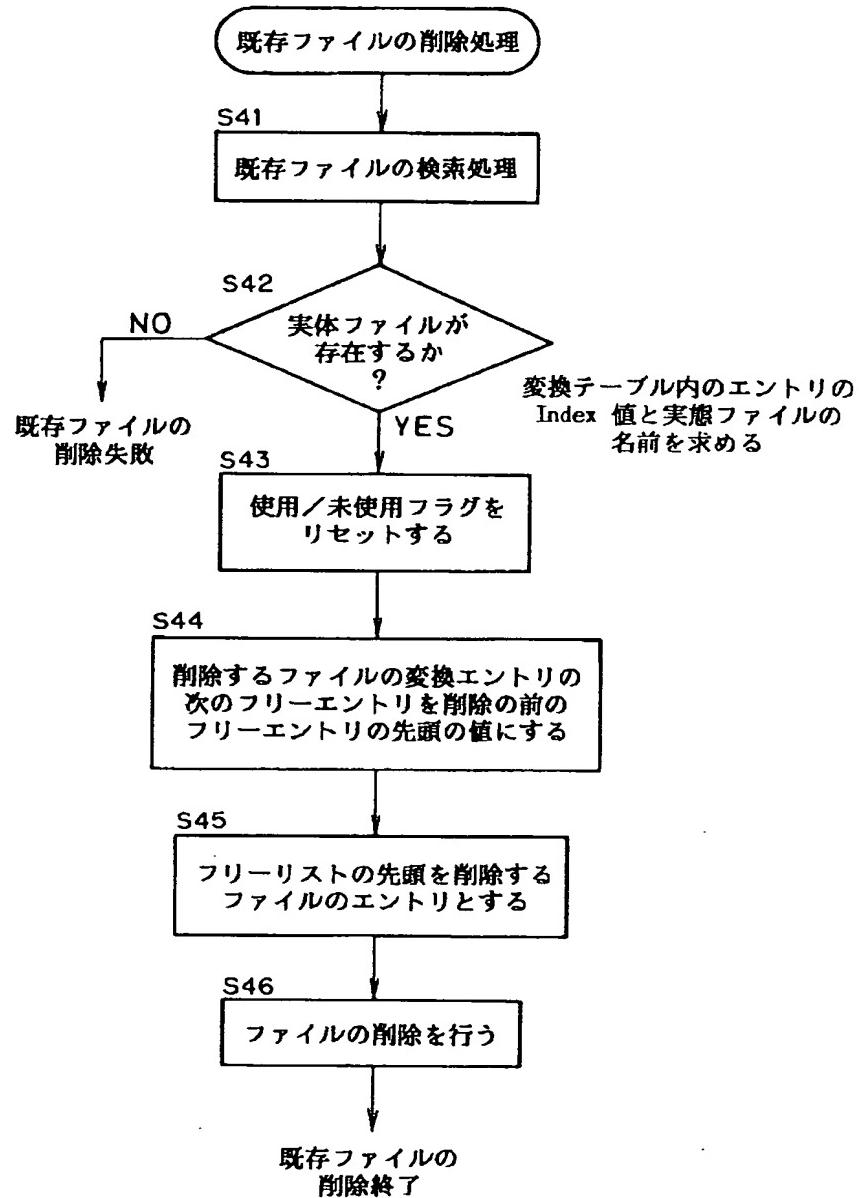
【図3】



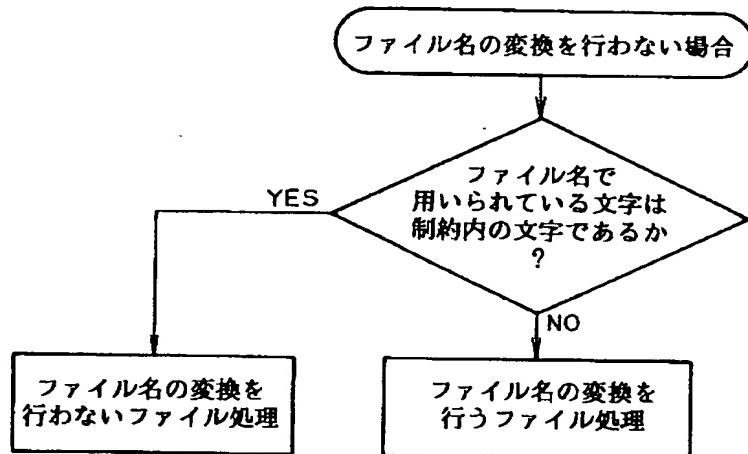
【図5】



【図7】



【図8】



【図9】

